[**Muhammed Nur Talha Kılıç**](https://medium.com/@mntalha.kilic?source=post_page-----7ca41c0ee4df--------------------------------)

[Lists](https://medium.com/@mntalha.kilic/lists?source=post_page-----7ca41c0ee4df--------------------------------)

[About](https://medium.com/@mntalha.kilic/about?source=post_page-----7ca41c0ee4df--------------------------------)



Muhammed Nur Talha Kılıç

Muhammed Nur Talha Kılıç



# Git & GitHub Beginners Guide-1

[Muhammed Nur Talha Kılıç](https://medium.com/@mntalha.kilic?source=post_page-----7ca41c0ee4df--------------------------------)

[Muhammed Nur Talha Kılıç](https://medium.com/@mntalha.kilic?source=post_page-----7ca41c0ee4df--------------------------------)

[1 day ago·6 min read](https://medium.com/@mntalha.kilic/git-github-beginners-guide-1-7ca41c0ee4df?source=post_page-----7ca41c0ee4df--------------------------------)



Photo by [Roman Synkevych](https://unsplash.com/@synkevych?utm_source=medium&utm_medium=referral) on [Unsplash](https://unsplash.com/?utm_source=medium&utm_medium=referral)

Inthis article, I want to focus on one of the popular version control systems, **git & GitHub**, which are required as the projects are enlarged. Of course, there are plenty of control systems but the most popular one is the “**git**”. In this essay, I am sure that it will be quite beneficial for people who want to learn from scratch as well as see various commands together.

## **What is git?**

It should begin like this **“git doesn’t mean GitHub”**. To dive deeper into this topic, git is a command-line tool that allows us to version control locally, on our computers.  
With this application, we add to our projects to be versionable

**As for GitHub,**

It is a web-based interface using git in essence. Thousands of users share their project here, by loading their repos to GitHub servers and opening as public, thus they have a chance to collaborate the other user all around the world.

**If you wonder How,**

The coders who have taken your projects and realized a bug send you a “pull request” with all changes they made. If it is appropriate for you, you can merge it so your code will be less problematic.

On the other hand, On the condition that we want to keep our repos in servers instead of locally, GitHub has quite useful features. In this manner, both we and another user can reach with remote access.

**One fundamental question may come into the minds,**

**What if the user doesn’t want to share his files?**

It is possible to save both “**public**” and “**private”**

* **Public**means that all can access your public repos
* **Private**means that only you and another contributor are allowed to access.

**What can be made with Git ?**

* To create a repository with our user names and email addresses.

Note : Before starting the projects, you are required to save your config variables, which are user.name and user.email in terminal line, or any other interface which you are using. Because all the changes you have made will be saved with the name of user.name and user.email

git config --global user.name "Talha Kilic"  
git config --global user.email mntalha.kilic@gmail.com

If you want to check whether there is any saved user.name , you can check like below.

git config user.name

* To save any point we define **(commit operation)** and provide us to return that location back, meaning that we can go back to the past.
* To be able to open access to different users.
* Have a chance to make modification by creating branches without harming to our master structure.
* To be able to see when the changes are made and by whom.
* To provide synchronization.

The following picture is taken from popular CS50 lecture slides,

https://miro.medium.com/max/30/1*co5_wZEwdsocNOBANFgMmg.png?q=20

<https://cdn.cs50.net/web/2020/spring/lectures/1/lecture1.pdf>

As it can be seen, the number of **a,b,c,d** from the remote server can be accessed by the other user who has pulled. The crucial point synchronization which means that all user with pull command access the same data, which allows us to work together.

**I also want to share a pretty good opportunity in GitHub,**

Student Developer Pack program provides many free features as long as you log in with student email.

https://miro.medium.com/max/30/1*UFMd-4zFQWdBBWd9QZtXKA.png?q=20

<https://education.github.com/pack>

Lots of websites is free with this pack such as Canvas, GitHub pro, etc.

**I definitely suggest you go and check“**[**https://education.github.com/pack**](https://education.github.com/pack)**".**

Anyway let’s come back to our topic,

It is possible to explain the Git life cycle like below.

https://miro.medium.com/max/30/1*dey0vBrkgJF1IXSw59b6Tw.png?q=20

[Git — Life Cycle (tutorialspoint.com)](https://www.tutorialspoint.com/git/git_life_cycle.htm)

The **repository**is your main directory, where you can put your project inside. It is also likely to save both locally or on a GitHub server.

Then with**clone operation**, you can take from the GitHub server and modify the original file.

You can make an **edit**, **add**and **move**operations.

If everything is alright, **add operation**upload the files you specify to the “**staging area”,** so when you change something inside the repo , you need to use **add command** to be able to commit.

After that you can point the project and create **SHA block**for coming back to the same point. With **commit operation**, name, date and email of the one who changes are saved. You can check from **logs**.

https://miro.medium.com/max/30/1*dey0vBrkgJF1IXSw59b6Tw.png?q=20

[Git — Life Cycle (tutorialspoint.com)](https://www.tutorialspoint.com/git/git_life_cycle.htm)

Up picture is a visualization of general opinion in Git.

After **commit operation**, if you want to upload the GitHub server, you need to use the push command.

Another important thing is the **pull operation**. If there is a change in a remote server, you can “**pull**” the all changes. Pull operation contains 2 commands which are **fetch** and **merge**.

https://miro.medium.com/max/30/1*Yk9nBPow30dcQP60CEM1oQ.png?q=20

<https://kheri.net/git-fetch-vs-git-pull/>

**Fetch**is download the file to the locale.

**Merge**combines the alteration with your local files. Providing that the same field is changed in both files, a conflict situation emerges. **You have to be careful with that.**

https://miro.medium.com/max/30/1*-E5Xir_7yVeca9b6kWjOKw.png?q=20

<https://cdn.cs50.net/web/2020/spring/lectures/1/lecture1.pdf>

## **Vital Points,**

* The Mainline is always **master**, which all changes should be merged with.
* In order to improve something, **branches**are so ideal because you can prevent any irreversible operation.

The next stop is **HEAD**, you can see the name of HEAD everywhere, it refers to the last point where you are.

After making a **branch** operation, you can merge it again.

https://miro.medium.com/max/30/1*t07oy5_czI4H2a4zUc4ozQ.png?q=20

<https://cdn.cs50.net/web/2020/spring/lectures/1/lecture1.pdf>

## **Git Commands,**

# **git init**

It is the first code black in order to initialize the project. It considers the whole file inside as a git repository.

# **git add demo.txt**

you can load “demo.txt” to **the staging area** with add command.

# **git add file1 file2 file3**

It is also possible to add multiple files.

# **git add .**

It adds all files in current directory.

# **git commit -m “Initial Commit”**

It creates a point which is SHA block for you to back the there. You can also comment about your general changes.

# **git status**

It provides general status. Red file implies that you didn't add . Green implies that you add these.

# **git log**

It shows as log format of commits and commits information until that time.

# **git pull**

It takes the changes from remote server.

# **git push**

It loads the changes from local to remote.

# **git checkout sha\_block**

It is used to go back previous commits.

*This is the final of the first writing about GitHub. In the subsequent article, we will dive into.*

I want to share my GitHub page as well. I would be glad if you follow

## [mntalha - Overview](https://github.com/mntalha/" \t "_blank)

### [Electronic Engineering MSc. mntalha has 19 repositories available. Follow their code on GitHub.](https://github.com/mntalha/" \t "_blank)

[github.com](https://github.com/mntalha/" \t "_blank)

If there are any incomprehensible points , you can always reach me from LinkedIn

## [Muhammed Nur Talha KILIÇ - İstanbul Teknik Üniversitesi - Istanbul, Istanbul, Turkey | LinkedIn](https://www.linkedin.com/in/mntalhakilic/" \t "_blank)

### [View Muhammed Nur Talha KILIÇ's profile on LinkedIn, the world's largest professional community. Muhammed Nur Talha has…](https://www.linkedin.com/in/mntalhakilic/" \t "_blank)

[www.linkedin.com](https://www.linkedin.com/in/mntalhakilic/" \t "_blank)

## **[Muhammed Nur Talha Kılıç](https://medium.com/@mntalha.kilic?source=post_sidebar--------------------------post_sidebar-----------)**

Istanbul Technical University — Electronics Engineering Master. “One day, in retrospect, the years of struggle will strike you as the most beautiful.” — S.Freud

* [Git](https://medium.com/tag/git)
* [Github](https://medium.com/tag/github)

## [**More from Muhammed Nur Talha Kılıç**](https://medium.com/@mntalha.kilic?source=follow_footer-----7ca41c0ee4df--------------------------------)

Istanbul Technical University — Electronics Engineering Master. “One day, in retrospect, the years of struggle will strike you as the most beautiful.” — S.Freud

[About](https://medium.com/about?autoplay=1&source=post_page-----7ca41c0ee4df--------------------------------)

[Write](https://medium.com/new-story?source=post_page-----7ca41c0ee4df--------------------------------)

[Help](https://help.medium.com/hc/en-us?source=post_page-----7ca41c0ee4df--------------------------------)

[Legal](https://policy.medium.com/medium-terms-of-service-9db0094a1e0f?source=post_page-----7ca41c0ee4df--------------------------------)